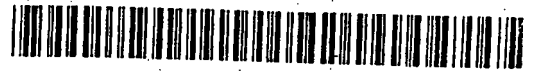


REF AL



①9 BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENT- UND  
MARKENAMT

⑫ Übersetzung der  
europäischen Patentschrift

⑤① Int. Cl. 7:  
G 06 F 3/12

⑨7 EP 0 733 965 B 1

⑩ DE 695 24 381 T 2

- ②① Deutsches Aktenzeichen: 695 24 381.0  
⑥⑥ Europäisches Aktenzeichen: 95 200 724.3  
⑧⑧ Europäischer Anmeldetag: 23. 3. 1995  
⑨7 Erstveröffentlichung durch das EPA: 25. 9. 1996  
⑨7 Veröffentlichungstag  
der Patenterteilung beim EPA: 5. 12. 2001  
④7 Veröffentlichungstag im Patentblatt: 14. 8. 2002

⑦③ Patentinhaber:  
Agfa-Gevaert N.V., Mortsel, BE

⑦④ Vertreter:  
derzeit kein Vertreter bestellt

⑧④ Benannte Vertragsstaaten:  
BE, DE, FR, GB, NL

⑦② Erfinder:  
Vennekens, Pierre, B-2640 Mortsel, BE

⑤④ Parallele Verarbeitung des Datenstroms der Seitenbeschreibungssprache

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

DE 695 24 381 T 2

DE 695 24 381 T 2

110202

BESCHREIBUNGTechnisches Gebiet

5 Die vorliegende Erfindung betrifft Einrichtungen und  
Verfahren zum Umsetzen eines Datenstroms in der  
Seitenbeschreibungssprache, der eines oder mehrere  
Bilder oder Seiten beschreibt, in eine Bitmap-  
10 Darstellung des Bildes zur Ausgabe auf Film, Papier,  
einem Videomonitor, einer Druckplatte usw. Die  
Erfindung kann in der Grafikindustrie, der  
Mikrographie, im Desktop-Publishing usw. angewandt  
werden. Die Erfindung betrifft insbesondere eine  
15 Mehrfach-Verarbeitungseinrichtung, die den Datenstrom  
in ein Zwischen- oder Bitmap-Format umsetzt.

Allgemeiner Stand der Technik

Die elektronische Wiedergabe durch Ausgabegeräte auf  
20 Hardcopy-Material erfordert eine schnelle Transfor-  
mation von Layouts, die gewöhnlich in einer bestimmten  
Seitenbeschreibungssprache (PDL) definiert werden, in  
eine Bitmap-Darstellung, der durch die Layouts  
beschriebenen Bilder. PDLs sind zum Beispiel PostScript  
25 (Warenzeichen der Adobe Inc.), AgfaScript (Warenzeichen  
der Agfa-Gevaert AG in Leverkusen), IPDS (Intelligent  
Printer Data Stream, Teil des Common Communications  
Support, definiert von IBM innerhalb des Kontexts der  
Systems Application Architecture). Eine Bitmap-  
30 Darstellung ist eine eindeutige Beziehung zwischen  
Bildpunkten, die von einem Ausgabegerät auf eine Druck-  
ausgabe geschrieben werden, und den elektronischen  
Signalen zur Veränderung der Dichte dieser Bildpunkte.  
Ein Bildpunkt eines Geräts ist das kleinste räumliche  
35 Element auf dem Hardcopy-Material, dessen Dichte durch  
das Ausgabegerät verändert werden kann. Wenn die Dichte  
des Bildpunkts eines Geräts zwei verschiedene Werte  
annehmen kann, z.B. schwarz und weiß, dann ist das

Ausgabegerät ein Zweipegelgerät, und in der Bitmap-Darstellung ist für jeden Bildpunkt des Geräts ein Bit notwendig, das eine Null oder eine Eins darstellt. Wenn das Ausgabegerät bis zu sechzehn Dichtepegel pro Bildpunkt des Geräts wiedergeben kann, dann sind in der Bitmap mindestens vier Bit pro Bildpunkt des Geräts erforderlich. Ausgabegeräte, die 256 Dichtepegel wiedergeben, erfordern acht Bit pro Bildpunkt des Geräts. Farbausgabegeräte erfordern gewöhnlich eine Bitmap pro Farbkomponente, z.B. Cyan, Magenta, Gelb und Schwarz. Die Größe der Bitmap bzw. Bitmaps ist also eine Funktion der Anzahl von Dichtepegeln, die von dem Ausgabegerät dargestellt werden, und der Anzahl von Farbkomponenten. Die Größe hängt außerdem von der räumlichen Auflösung der Bildpunkte des Geräts ab. Die räumliche Auflösung des Geräts wird in Punkten pro Zoll oder Bildpunkten pro Millimeter ausgedrückt. Je höher die Auflösung ist, desto größer wird die Bitmap. Außerdem ist die Gesamtgröße der Druckausgabe entscheidend für die in der Bitmap erforderliche Anzahl von Speicherelementen. Bei einem Zweipegel-A4-Drucker mißt die Druckausgabe 297 mm mal 210 mm, mit einer Auflösung von 400 Punkten pro Zoll oder etwa 16 Bitpunkten pro mm, und eine Bitmap erfordert fast 15 Megabit ( $15 \text{ mal } 2^{20} \text{ Bit}$ ). Da zunehmend qualitativ hochwertige Wiedergaben verlangt werden, hat die räumliche Auflösung bis auf 600 Punkte pro Zoll, die Anzahl von Farbkomponenten auf vier, die Anzahl von Dichtepegeln pro Farbkomponenten auf sechzehn und die Größe auf A3 (420 mm mal 297 mm) zugenommen. Dadurch erhöht sich die erforderliche Menge an Speicher bis auf etwa 1062 Megabit. Diese große Speichermenge ist nicht nur für die Speicherkapazität problematisch, sondern auch für die Erzeugung dieser großen Datenmenge in der Bitmap, durch die das Bild auf der Druckausgabe dargestellt wird. Das Ausgabegerät setzt die Bitmap-Darstellung in ein sichtbares Bild auf einer Druckausgabe oder auf einem Anzeigemonitor um.

Die Transformation eines PDL-Datenstroms in eine Bitmap-Darstellung geschieht in einem Bildbearbeitungsprozessor (RIP - Raster Image Processor). Diese  
5 Einrichtung empfängt einen PDL-Datenstrom, interpretiert seinen Inhalt und erzeugt eine Bitmap-Darstellung, die für das Ausgabegerät geeignet ist, d.h. die räumliche Auflösung, die Anzahl von Dichtepegeln, die Anzahl von Farbkomponenten und die  
10 Größe der Bitmap entspricht den Kenngrößen des Ausgabegeräts.

Obwohl die Auflösung drastisch zugenommen hat, und deshalb auch die Datentransfermenge zwischen dem RIP  
15 und der Anzeige- oder Wiedergabeeinrichtung sowie zum RIP selbst, ist der RIP bei der Übersetzung des PDL-Datenstroms in eine Bitmap zum Engpaß geworden. Diese Entwicklung ist auf den höheren Durchsatz der derzeitigen Übertragungsmittel zurückzuführen. Der  
20 Durchsatz eines Übersetzungsvorgangs in einem RIP kann über die Datenmenge in dem PDL-Datenstrom pro Zeiteinheit definiert werden, die durch das RIP-System verarbeitet werden kann. In der Vergangenheit wurden verschiedene Maßnahmen zur Vergrößerung des Durchsatzes  
25 getroffen:

- 1) Der Durchsatz des Hardware-Prozessors, auf dem der Übersetzungsprozeß abläuft, kann durch Verwendung eines schnelleren Prozessors oder  
30 eigener Hardware vergrößert werden;
- 2) Programmiertechniken oder -sprachen können angewandt werden, um die Gesamtsystemleistung zu verbessern;
- 3) Der Umsetzungsprozeß kann sequenziell in  
35 Teilprozesse aufgeteilt werden, die auf verschiedenen Prozessoren ausgeführt werden.

Der Nachteil der Verwendung eigener Hardware, z.B. zur Erzeugung einer Bitmap, ist die schlechte Wirtschaftlichkeit. Wenn es zu Änderungen der Anforderungen an Funktion, Leistung oder Vermarktung kommt, kann eine

5 hohe Investition in Hardware-Aktualisierung notwendig sein. Ein System, das auf standardmäßigen Hardware-Komponenten und konfigurierbaren Softwaremodulen basiert, gestattet eine höhere Flexibilität, um schnell auf sich ändernde Anforderungen zu reagieren. Somit

10 besteht ein großer Bedarf, den Durchsatz eines Bildbearbeitungsprozessors mit kostengünstigen fertigen Hardware-Komponenten und konfigurierbaren Software-techniken zu vergrößern. Auch die WO-A 94 11805 betrifft die Verarbeitung von PDL-Daten, es wird jedoch

15 keine parallele Verarbeitung wie in der vorliegenden Anmeldung besprochen.

#### Aufgaben der Erfindung

20 Eine erste Aufgabe der Erfindung besteht deshalb darin, ein Verfahren zur Verbesserung der Leistung eines Bildbearbeitungsprozessors bereitzustellen.

Eine weitere Aufgabe der Erfindung ist die

25 Bereitstellung eines Verfahrens, bei dem mehrere Prozessoren gleichzeitig arbeiten können.

Noch eine weitere Aufgabe der Erfindung ist die Bereitstellung eines Verfahrens, so daß die Leistung

30 nahezu eine lineare Funktion der Anzahl von parallel arbeitenden Prozessoren ist.

Eine besondere Aufgabe der Erfindung besteht darin, daß trotz der parallelen Verarbeitung die korrekte

35 Seitenfolge, die zu dem Anzeige- oder Wiedergabegerät oder dem zugrundeliegenden System gesendet wurde, aufrechterhalten wird.

Weitere Aufgaben und Vorteile der Erfindung werden aus der folgenden Beschreibung ersichtlich.

#### Kurze Darstellung der Erfindung

5 Die oben angeführten Aufgaben können durch ein Verfahren mit den in Anspruch 1 dargelegten spezifischen Merkmalen realisiert werden. Ein Bildwiedergabesystem und ein Bildbearbeitungsprozessor,  
10 die das erfindungsgemäße Verfahren verwenden können, werden ebenfalls hier beschrieben und in dem unabhängigen Anspruch 13 beansprucht.

Spezifische Merkmale für bevorzugte Ausführungsformen  
15 der Erfindung werden in den abhängigen Ansprüchen dargelegt.

Der PDL-Datenstrom wird gewöhnlich in einen Übergangs-Datenstrom umgesetzt, der sich für eine Umsetzung in  
20 eine Bitmap-Darstellung zur Wiedergabe des durch den PDL-Datenstrom beschriebenen Bildes eignet.

Die Erzeugung unabhängiger PDL-Datenstromsegmente hat den Vorteil, daß die Segmente gleichzeitig verarbeitet  
25 werden können und die Verarbeitungslast über verschiedene Teilprozesse verteilt werden kann, die parallel auf mehreren Prozessoren ablaufen. Bei einer bevorzugten Ausführungsform wird aus Steuerbefehlen ein Übersetzungszustand aufgebaut und durch die  
30 Teilprozesse aufrechterhalten, die die Segmente in Übergangs-Datenstromteile übersetzen. Dies hat den Vorteil, daß zwischen Teilprozessen keine Kommunikation erforderlich ist. Bei einer anderen Ausführungsform kann der Übersetzungszustand in einem zentralen Prozeß  
35 aufgebaut werden. Jeder Teilprozeß muß von diesem zentralen Prozeß den korrekten Übersetzungszustand erhalten. Bei einer besonders bevorzugten Ausführungsform wird die Übermittlung von

Steuerbefehlen reduziert. Ferner wird die Verarbeitung erweiterter Steuerbefehle, wie unten definiert, vorzugsweise von einem Teilprozeß, möglicherweise für jeden Befehl unterschiedlich, ausgeführt. Ein zentraler Hauptprozeß wäre ein Engpaß, wenn er alle solchen Befehle verarbeiten müßte. Durch Verteilen erweiterter Steuerbefehle auf verschiedene Teilprozesse können diese Befehle darüber hinaus außerdem parallel oder gleichzeitig verarbeitet werden.

10

Ein PDL-Datenstrom ist, wie bereits erwähnt, eine Folge elektronischer Signale, die in einer bestimmten Seitenbeschreibungssprache auf höherer Ebene beschreiben, wie ein Bild auf einer Druckausgabe oder auf einer Videoanzeigeeinheit aussehen muß. Das Bild kann nur aus Text bestehen, wie gedruckte Seiten. Das Bild kann außerdem Grafiken, Bilder, Halbtonbilder usw. enthalten. Beispiele für PDL sind PostScript, AgfaScript, IPDS. Ein Übergangs-Datenstrom ist eine Folge elektrischer Signale, die dasselbe Bild, aber auf einer niedrigeren Ebene beschreiben. Ein typisches Beispiel ist eine Anzeigeliste, die ein Bild als eine Kombination von Trapezen beschreibt. Die Umsetzung aus dem Übergangs-Datenstromformat in eine Bitmap-Darstellung kann dann auf effiziente Weise durchgeführt werden. Gemäß dem Verfahren der vorliegenden Erfindung kann der Übergangs-Datenstrom ebenfalls eine Bitmap-Darstellung des auf dem Ausgabemedium wiederzugebenden Bildes sein. Dementsprechend ist kein zusätzlicher Umsetzungsschritt aus dem Übergangs-Datenstromformat in eine Bitmap-Darstellung erforderlich. Der Unterschied zwischen Datenbefehlen und Steuerbefehlen wird in der nachfolgenden ausführlichen Beschreibung weiter besprochen.

35

Ein unabhängiges PDL-Datenstromsegment ist ein Teil des vollständigen PDL-Datenstroms. Alle Datenbefehle aus dem PDL-Datenstrom können in derselben Sequenz in den

11.02.02

Segmenten gefunden werden. Alle Datenbefehle, die einer Seite oder einem Bereich entsprechen, werden immer in nur einem Segment gruppiert. Ein Bereich ist ein disjunktiver Teil eines physischen Mediums. Ein Segment  
5 beschreibt also das vollständige Bild, das auf mindestens einem solchen Bereich wiedergegeben werden soll. Gemäß dem Verfahren der vorliegenden Erfindung wird also jeder Datenbefehl aus dem PDL-Datenstrom nur einem Segment zugewiesen. Im Gegensatz zu Steuerbe-  
10 fehlen, die über verschiedene Segmente hinweg vervielfältigt werden können, wird ein spezifischer Datenbefehl nicht über mehrere Segmente hinweg vervielfältigt oder verteilt und befindet sich nur in einem Segment. Ein Segment enthält gewöhnlich mehrere Daten-  
15 befehle, die die Beschreibung eines Bereichs bilden. Darüber hinaus kann ein Segment Datenbefehle enthalten, die mehr als einem Bereich entsprechen.

Im Gegensatz zu Datenbefehlen können zusätzliche  
20 Steuerbefehle hinzugefügt werden, damit jedes Segment unabhängig wird. Gemäß dem Verfahren der vorliegenden Erfindung wird mindestens ein Steuerbefehl einmal oder mehrfach vervielfältigt und in zwei oder mehreren verschiedenen Segmenten gesendet. Außerdem ist es  
25 möglich, daß einige überflüssige Steuerbefehle aus den erzeugten Segmenten entfernt werden, um die Leistung zu steigern. Ein solches Segment beschreibt einen oder mehrere Bereiche oder eine oder mehrere Seiten. Wie bereits erwähnt, ist ein Bereich ein disjunktiver Teil  
30 eines physischen Mediums. Im einfachsten Fall beschreibt jedes Segment ein Bild, das auf einer einzelnen Seite gedruckt werden soll. Bei der Mikrographie werden verschiedene "Seiten" auf einem einzigen physischen Mikrofilm angeordnet. In diesem  
35 Fall kann das vollständige Bild auf einem Mikrofilm durch mehrere unabhängige PDL-Datenstromsegmente beschrieben werden. Außerdem ist es möglich, daß eine physische Seite aus zwei rechteckigen Textblöcken und



11.02.02

- einem Bereich um die Blöcke herum zusammengesetzt ist. Jeder einzelne Textblock kann durch ein unabhängiges PDL-Datenstromsegment beschrieben werden, das aus einem Textbild zusammengesetzt ist, und der Bereich um diese
- 5 Blöcke kann ein anderes Bild darstellen, das durch ein unabhängiges PDL-Datenstromsegment beschrieben wird, das Grafiken, Referenzen, Positionsmarkierungen usw. umfaßt. Bei einer anderen Ausführungsform kann eine physische Seite in aufeinanderfolgende angrenzende,
- 10 aber disjunktive Bänder aufgeteilt werden. Datenbefehle und Steuerbefehle, die sich auf das Bild in jedem einzelnen Band auswirken, können in einem unabhängigen PDL-Datenstromsegment angeordnet werden.
- 15 Die Transformation des PDL-Datenstroms in unabhängige PDL-Datenstromsegmente geschieht vorzugsweise durch einen Hauptprozeß. Die Umsetzung einzelner unabhängiger PDL-Datenstromsegmente in Übergangs-Datenstromteile wird vorzugsweise von einem oder mehreren Teilprozessen
- 20 durchgeführt. Der Hauptprozeß und die mehreren Teilprozesse können gleichzeitig auf einem oder mehreren Verarbeitungssystemen ablaufen. Die Anzahl von Prozessen kann größer oder gleich der Anzahl von Prozessoren sein. Dementsprechend kann die Übersetzung
- 25 unabhängiger PDL-Datenstromsegmente parallel auf verschiedenen Prozessoren ablaufen. Durch die Wahl der Menge von Teilprozessen, der Menge von Prozessorsystemen und der Verteilung der Teilprozesse über die Prozessorsysteme kann der Grad der Parallelisierung
- 30 abgestimmt werden. Bei Verwendung des erfindungsgemäßen Verfahrens ist der Durchsatz nahezu eine lineare Funktion der Anzahl von Prozessorsystemen.

#### Ausführliche Beschreibung der Erfindung

35

Die Erfindung wird im folgenden anhand von Beispielen unter Bezugnahme auf die beigefügte Zeichnung beschrieben. Es zeigt:

11.02.03

FIG. 1 eine spezifische Ausführungsform zur Ausführung des Verfahrens gemäß der vorliegenden Erfindung.

Mit Bezug auf FIG. 1 enthält ein PDL-Datenstrom 31 hauptsächlich Befehle und wahlweise z.B. Kommentare. Die Kommentare müssen gewöhnlich nicht weiter verarbeitet werden. Die Befehle können in Datenbefehle und Steuerbefehle aufgeteilt werden. Die Steuerbefehle können weiter in erweiterte Steuerbefehle und reduzierte Steuerbefehle aufgeteilt werden. Die Unterscheidung zwischen erweiterten und reduzierten Steuerbefehlen kann auf der Grundlage ihrer Semantik, aber auch auf der Grundlage ihrer Größe erfolgen.

15 Datenbefehle beschreiben, welche Daten, wie zum Beispiel Text, Grafiken und Bilder, wiedergegeben werden müssen. Der Inhalt eines Datenbefehls kann die wiederzugebenden Daten vollständig beschreiben, der Datenbefehl umfaßt jedoch gewöhnlich eine implizite oder explizite Referenz auf erweiterte Datenstrukturen. Diese werden durch erweiterte Steuerbefehle definiert oder liegen permanent in dem Bildbearbeitungsprozessorsystem vor. Typische erweiterte Datenstrukturen sind verankerte oder heruntergeladene Schriftsatzbeschreibungen, Seitenteile, Formulare und Overlays.

Während Datenbefehle beschreiben, welche Daten wiedergegeben werden müssen, definieren Steuerbefehle, wie die Daten wiedergegeben werden müssen. Zur Verarbeitung eines Datenbefehls müssen die Teilprozesse 36, 37, die PDL-Datenstromsegmente in Übergangs-Datenstromsegmente umsetzen, einen Übersetzungszustand behandeln, der diesen Datenbefehl betrifft. Der Übersetzungszustand wird aus den vorherigen Steuerbefehlen aufgebaut. Bei bestimmten PDLs wirken lokale Steuerbefehle nur einmal und werden von einem lokalen Vorgabe-Steuerbefehl ersetzt, wenn der erforderliche lokale Befehl nicht vorliegt. Lokale Steuerbefehle tragen nicht zu dem

11.02.02

Übersetzungszustand bei. Solche PDLs definieren globale Steuerbefehle als wirksam, bis ihre Wirkung durch einen anderen globalen Steuerbefehl aufgehoben wird. Ihre Wirkung bleibt über nachfolgende Bereiche oder Seiten hinweg gültig, und dementsprechend tragen globale Steuerbefehle zu dem Übersetzungszustand bei.

Die meisten PDL-Datenströme beschreiben nachfolgende Bereiche bereichs- oder seitenunabhängig. Es muß aber eine klare Trennung von Datenbefehlen, die zu einem Bereich gehören, möglich sein. Bei der Sprache IPDS wird zum Beispiel der Beginn von Datenbefehlen, die eine Seite betreffen, durch den Befehl "BeginPage" angegeben. Der letzte Datenbefehl, der mit einer Seite zusammenhängt, wird durch den Befehl "EndPage" angegeben. Die Teilprozesse 36, 37, die die Datenbefehle in Übergangs-Datenstromteile übersetzen, müssen über eine vollständige Kenntnis des Übersetzungszustands verfügen, was entscheidend für die korrekte Übersetzung der Datenbefehle aus dem PDL-Datenstrom in den Übergangs-Datenstrom ist. Der Übersetzungszustand ist aus allen vorherigen Steuerbefehlen zusammengesetzt, einschließlich der Steuerbefehle, die den Übersetzungszustand vorheriger Bereiche oder Seiten definieren. Jeder Teilprozeß muß entweder den Übersetzungszustand über verschiedene unabhängige PDL-Datenstromsegmente hinweg mitverfolgen oder muß durch den Inhalt jedes unabhängigen PDL-Datenstromsegments eine vollständige Kenntnis des Übersetzungszustands für das aktuelle unabhängige PDL-Datenstromsegment erhalten. Die erste Alternative ist nur schwer zu realisieren, da aufgrund der Vielzahl von Teilprozessen jeder Teilprozeß nur einen Teil der vollständigen Menge unabhängiger PDL-Datenstromsegmente übersetzt.

35

Erweiterte Steuerbefehle können erweiterte Datenstrukturen gemäß der obigen Beschreibung definieren und löschen. Obwohl das Löschen einer erweiterten

- Datenstruktur durch einen kurzen Befehl angewiesen werden kann, wird ein solcher Löschbefehl vorzugsweise ebenfalls als ein erweiterter Steuerbefehl behandelt. Diese erweiterten Datenstrukturen werden vorzugsweise vorübergehend oder permanent auf einem Speichergerät 5 46, wie zum Beispiel einer Festplatte, einem anderen Magnetspeicher oder einem elektronischen Speichermedium, wie zum Beispiel RAM, ROM, PROM, EPROM, EEPROM usw. gespeichert. Vorzugsweise wird die Verwaltung 10 erweiterter Datenstrukturen einem zentralen Speicherprozeß 45 zugewiesen. Als Alternative kann diese Aufgabe durch den Hauptprozeß 32 durchgeführt werden. Erweiterte Datenstrukturen können zu diesem zentralen Speicherprozeß 45, der die Strukturen in ein 15 Datenformat umsetzt, das sich für das Speichermedium eignet, gesendet oder von diesem abgerufen werden. Es können mehrere Teilprozesse gleichzeitig erweiterte Datenstrukturen senden und abrufen.
- 20 Mit Bezug auf FIG. 1 empfängt der Hauptprozeß 32 den PDL-Datenstrom 31 gewöhnlich von einem (nicht gezeigten) externen Gerät, auf dem eine Bedienungsperson interaktiv die auf der Druckausgabe wiederzugebenden Bilder spezifiziert hat. Der PDL- 25 Datenstrom 31 kann außerdem durch ein Drucker-Spooler-Programm erzeugt werden, das z.B. auf Mikrofilm zu druckende Account-Informationen erzeugt. Gewöhnlich kann der Hauptprozeß 32 auch Fehler, Ausnahmen oder Statusinformationen auf Anforderung zu dem externen 30 Gerät zurücksenden.

Bei einer bevorzugten Ausführungsform erzeugt der Hauptprozeß 32 fehlerfreie unabhängige PDL-Datenstromsegmente, d.h. die Teilprozesse 36, 37 stoßen auf keine 35 Ausnahmen, für die eine Kommunikation mit dem Hauptprozeß 32 notwendig ist. Dies bedeutet außerdem, daß Fehler, Unzulänglichkeiten oder Vorgabewerte, die in

dem PDL-Datenstrom auftreten, vorzugsweise durch den Hauptprozeß 32 abgewickelt werden.

Bei einer besonders bevorzugten Ausführungsform führt  
5 der Hauptprozeß 32 eine beschränkte Syntaxprüfung durch, um festzustellen, ob die ankommenden Befehle interpretiert werden können. Dazu kann eine Prüfung der Länge der Befehle, eine Prüfung von Kopfteilen usw. gehören. In diesem Fall muß jeder Teilprozeß Semantik-  
10 prüfungen durchführen und dadurch bestimmen, ob die Datenbefehle tatsächlich in auf dem physischen Ausgabe-  
medium wiederzugebende Bildteile umgesetzt werden können. Dazu gehört in der Regel eine Prüfung, ob die referenzierten Fonts oder Fontparameter definiert  
15 worden sind usw. Die Verteilung dieser Semantikprüfungen auf die Teilprozesse 36, 37 hat den Vorteil, daß der Hauptprozeß 32 entlastet wird. Fehlerantworten aus dem Teilprozeß 36, 37 zum Hauptprozeß 32 werden vorzugsweise über denselben Weg  
20 gesendet wie Steuerbefehlsantworten 44 (siehe unten).

Der Hauptprozeß 32 analysiert den PDL-Datenstrom 31 und insbesondere die Steuerbefehle. Die Hauptaufgabe des Hauptprozesses 32 besteht darin, unabhängige PDL-  
25 Datenstromsegmente 33 zu erzeugen, so daß jedes Segment unabhängig von jedem anderen Segment verarbeitet werden kann. Diese Unabhängigkeit garantiert, daß eine parallele Verarbeitung, auch durch verschiedene Verarbeitungssysteme, machbar ist. Dazu kann man in jedes  
30 Segment eine vollständige Übersetzungszustandsbeschreibung einfügen, die die Datenbefehle in dem Segment betrifft. Da jedes unabhängige PDL-Datenstromsegment genug Datenbefehle enthält, um zu beschreiben, welche Bilder auf dem Bereich oder der Seite wiedergegeben  
35 werden müssen, und eine vollständige Menge von Steuerbefehlen, um anzuweisen, wie die Datenbefehle interpretiert werden müssen, können die unabhängigen PDL-Datenstromsegmente unabhängig voneinander verarbei-

11.02.02

tet werden. Vorzugsweise wird jedes unabhängige PDL-Datenstromsegment oder eine Referenz auf dieses Segment, die häufig durch einen Speicheradressenzeiger realisiert wird, durch den Hauptprozeß 32 in eine FIFO-Warteschlange (first in, first out) 34 eingereiht. Die FIFO-Organisation stellt sicher, daß die unabhängigen PDL-Datenstromsegmente von den Teilprozessen 36, 37 in derselben Reihenfolge gewählt werden, in der diese unabhängigen PDL-Datenstromsegmente von dem Hauptprozeß 32 erzeugt werden. Ein oder mehrere Teilprozesse 36, 37 können dann auf die FIFO-Warteschlange zugreifen und ein unabhängiges PDL-Datenstromsegment 35 zur Verarbeitung zuteilen. Da jedes unabhängige PDL-Datenstromsegment selbstständig ist, ist es gleichgültig, welcher Teilprozeß welches unabhängige PDL-Datenstromsegment entnimmt. Immer wenn ein Teilprozeß mit einem unabhängigen PDL-Datenstromsegment fertig ist, kann er der FIFO-Warteschlange ein weiteres unabhängiges PDL-Datenstromsegment entnehmen. Somit wird nicht ein spezifisches unabhängiges PDL-Datenstromsegment einem spezifischen Teilprozeß zugewiesen, sondern jeder Teilprozeß wählt das nächste unabhängige PDL-Datenstromsegment, das zur Verarbeitung verfügbar ist, aus der FIFO-Warteschlange. Dementsprechend wird die Last der Teilprozesse 36, 37 automatisch ausgeglichen und die Übersetzungsarbeit, die für verschiedene unabhängige PDL-Datenstromsegmente sehr unterschiedlich sein kann, gleichmäßig über die Teilprozesse 36, 37 verteilt. Da jedes unabhängige PDL-Datenstromsegment selbstständig ist, ist keine zusätzliche Kommunikation wechselseitig zwischen den Teilprozessen 36, 37 oder zwischen dem Hauptprozeß 32 und jedem Teilprozeß notwendig. Dies ist ein großer Vorteil, wenn die verschiedenen Teilprozesse 36, 37 über verschiedene Prozessoren verteilt sind. Die wechselseitige Kommunikation von Daten, z.B. unabhängigen PDL-Datenstromsegmenten, zwischen den Prozessoren erfolgt:

11.02.02

- über eine gemeinsame Busstruktur, z.B. einen VME-Bus;
- über Punkt-zu-Punkt-Strecken, z.B. RS232-C seriell, Parallel Centronics, usw.;
- 5 - über Netzwerkverbindungen, z.B. Ethernet, Ethertalk, usw.

Über diese Verbindung muß nur das vollständige unabhängige PDL-Datenstromsegment übertragen werden.

10 Vorzugsweise besitzt jeder Prozessor einen unabhängigen Speicher, der erforderlich ist, um die Daten in einem unabhängigen PDL-Datenstromsegment, den Übergangs-Datenstromteil 38, der aus dem unabhängigen PDL-Datenstromsegment 35 erzeugt wird, und bestimmte lokale

15 Datenstrukturen, die für die Übersetzung des unabhängigen PDL-Datenstromsegments in den Übergangs-Datenstrom erforderlich sind, zu halten.

Der Hauptprozeß 32 und die Teilprozesse 36, 37 können

20 frei über die verfügbaren Prozessorsysteme verteilt werden. Wenn zwei Prozessorsysteme A und B verfügbar sind, könnte der Hauptprozeß 32 auf dem Prozessorsystem A ablaufen, und ein Teilprozeß könnte auf dem Prozessorsystem B ablaufen. Wenn ein zweiter Teilprozeß

25 installiert wäre, könnte dieser auf dem Prozessorsystem A gleichzeitig mit dem Hauptprozeß 32 ablaufen. Wenn ein dritter Teilprozeß installiert wäre, könnte dieser auf dem Prozessorsystem B gleichzeitig mit dem ersten Teilprozeß ablaufen. Bei einer bevorzugten

30 Ausführungsform ist die Anzahl von Prozessorsystemen nicht kleiner als die Anzahl von Teilprozessen. Wenn zwei Prozesse auf einem Prozessorsystem ablaufen, nehmen sie vorzugsweise jeweils eine Partition des unabhängigen Speichers ein. Die Übertragung eines

35 unabhängigen PDL-Datenstromsegments von dem Hauptprozeß 32 zu einem Teilprozeß auf demselben Prozessorsystem kann durch eine Speicherkopie des unabhängigen PDL-Datenstromsegments aus der Hauptprozeßpartition zu der

Teilprozeßpartition erfolgen. Als Alternative können die Speicherblöcke, die das unabhängige PDL-Datenstromsegment enthalten, aus der Hauptprozeßpartition freigegeben und, vorzugsweise durch den Teilprozeß, der Teilprozeßpartition zugewiesen werden. Solche Blöcke können der Hauptprozeßpartition wieder zugewiesen werden, sobald der Teilprozeß mit der Verarbeitung des unabhängigen PDL-Datenstromsegments fertig ist. Auch wenn der Hauptprozeß 32 und der Teilprozeß nicht auf demselben Prozessorsystem ablaufen, ist es möglich, dem Teilprozeß einen Zeiger auf die Speicherstelle zu geben, an der das unabhängige PDL-Datenstromsegment verankert ist. Der Teilprozeß 36 oder 37 kann dann über den Bus, die Verbindung oder das Netzwerk, wodurch die beiden Prozessorsysteme verbunden sind, auf die Daten in dem unabhängigen PDL-Datenstromsegment zugreifen.

Bei der oben beschriebenen ersten Ausführungsform baut der Hauptprozeß 32 aus Steuerbefehlen, die in dem PDL-Datenstrom vorliegen, einen Übersetzungszustand auf. Jeder Teilprozeß erhält eine vollständige Übersetzungszustandsbeschreibung durch Steuerbefehle in einem Kopfteil jedes unabhängigen PDL-Datenstromsegments. Die Übertragung von Steuerbefehlen, die den aktuellen Übersetzungszustand vollständig beschreiben, kann den folgenden Teilen eine große Last auferlegen:

- dem Hauptprozeß 32, der die Steuerbefehle erzeugt;
- 30 - dem Übertragungssystem zur Übermittlung der Steuerbefehle von dem Hauptprozeß 32 zu dem Teilprozeß; und
- dem Teilprozeß für die Analyse der Steuerbefehle und den Aufbau eines aktuellen Zustands, der für die Verarbeitung der Datenbefehle in dem vorliegenden Segment geeignet ist.



110202

Eine Alternative könnte darin bestehen, den Übersetzungszustand zentral zu halten.

Bei einer bevorzugten Ausführungsform baut jeder Teilprozeß lokal einen Teilprozeß-Übersetzungszustand auf und behält nach der Verarbeitung eines unabhängigen PDL-Datenstromsegments im Kontext des Teilprozeß-Übersetzungszustands die Zustandsinformationen als Startpunkt zur Verwendung während der Verarbeitung des nächsten unabhängigen PDL-Datenstromsegments bei. Vorzugsweise enthält der Kopfteil des unabhängigen PDL-Datenstromsegments Steuerbefehle, um den Teilprozeß-Übersetzungszustand für die korrekte Verarbeitung der Datenbefehle in dem unabhängigen PDL-Datenstromsegment geeignet einzustellen. Somit muß der Hauptprozeß 32 nur Änderungen zwischen einem aktuellen Hauptübersetzungszustand und dem Teilprozeß-Übersetzungszustand übermitteln. Bisher gibt es jedoch keinen Mechanismus, durch den der Hauptprozeß 32 den Teilprozeß-Übersetzungszustand erfahren kann, was erforderlich ist, um die Änderungen zu übermitteln. Außerdem kann jeder Teilprozeß einen anderen Teilprozeß-Übersetzungszustand aufweisen, und der Hauptprozeß 32 entscheidet nicht, durch welchen Teilprozeß ein spezifisches unabhängiges PDL-Datenstromsegment verarbeitet wird.

Bei einer bevorzugten Ausführungsform gibt daher jeder Teilprozeß, der seinen Teilprozeß-Übersetzungszustand gemäß Steuerbefehlen verändert, diese Änderung als Antwort mittels einer Steuerbefehlantwort 44 an den Hauptprozeß 32 zurück. Der Hauptprozeß 32 sammelt diese Antworten für jeden Teilprozeß und verfolgt ein gemeinsames Übersetzungszustands-Level. Das gemeinsame Übersetzungszustands-Level ist definiert als der Übersetzungszustand, der aus denjenigen Steuerbefehlen aufgebaut wird, die von allen Teilprozessen verarbeitet und beantwortet wurden. Gemäß dieser Definition hat

jeder Teilprozeß einen Teilprozeß-Übersetzungszustand erreicht, der mindestens so neu ist wie das gemeinsame Übersetzungszustands-Level. Deshalb kann der Hauptprozeß 32 Zustandsänderungen sicher auf der Grundlage des Unterschieds zwischen dem aktuellen Hauptübersetzungszustand und dem gemeinsamen Übersetzungszustands-Level durchführen. Diese Zustandsänderungen zeigen an, welche Steuerbefehle in dem nächsten unabhängigen PDL-Datenstromsegment, das erzeugt wird, übertragen werden müssen. Wenn mehr als ein Teilprozeß die unabhängigen PDL-Datenstromsegmente interpretiert, ist es sehr wahrscheinlich, daß mindestens ein Teilprozeß einen Teilprozeß-Übersetzungszustand erreicht hat, der neuer ist als das gemeinsame Übersetzungszustands-Level. Wenn ein solcher Teilprozeß ein nächstes unabhängiges PDL-Datenstromsegment zur Verarbeitung erhält, enthält dieses unabhängige PDL-Datenstromsegment Steuerbefehle, die bereits von dem Teilprozeß verarbeitet wurden. Der Teilprozeß verwirft vorzugsweise diese überflüssigen Steuerbefehle, führt keine Zustandsänderung durch und gibt keine Antwort an den Hauptprozeß 32 aus, da die Zustandsänderung und die Antwort bereits erzeugt wurden. Somit ist es klar, daß jeder Teilprozeß nur einmal auf jeden nicht überflüssigen Steuerbefehl antwortet. Ob ein Steuerbefehl überflüssig ist oder nicht, hängt von dem Übersetzungszustand des Teilprozesses ab. In diesem Fall kann der Hauptprozeß 32 mitverfolgen, wie oft auf einen spezifischen Steuerbefehl geantwortet wurde. Wenn diese Anzahl für einen spezifischen Steuerbefehl die Anzahl von Teilprozessen in dem System erreicht, aktualisiert dieser spezifische Steuerbefehl den gemeinsamen Übersetzungszustands-Level. Auf diese Weise muß der Hauptprozeß 32 nicht den Teilprozeß-Übersetzungszustand mitverfolgen, der von jedem Teilprozeß einzeln zurückgegeben wird. Dieser Prozeß reduziert auf der Grundlage der Antwort auf Steuerbefehle drastisch die

Menge von Steuerbefehlen, die von dem Hauptprozeß 32 in den unabhängigen PDL-Datenstromsegmenten ausgegeben wird, und wird deshalb als Steuerbefehlreduktion bezeichnet.

5

Eine weitere Steuerbefehlreduktion wird vorzugsweise durch den folgenden Prozeß erzielt. Der Hauptprozeß 32 analysiert nicht das Vorliegen jedes Steuerbefehls in dem PDL-Datenstrom, sondern seinen Wert. Wenn aus dem Inhalt des Steuerbefehls klar ist, daß er den Hauptübersetzungszustand nicht ändert, wird der Steuerbefehl in kein unabhängiges PDL-Datenstromsegment eingefügt. Auf diese Weise reduziert der Hauptprozeß 32 die Menge von Steuerbefehlen, ohne die Unabhängigkeit der PDL-Datenstromsegmente zu beeinträchtigen. Diese weitere Steuerbefehlreduktion führt zu einem zusätzlichen Verarbeitungsschritt für den Hauptprozeß 32. Abhängig von der Last dieser zusätzlichen Steuerbefehlverarbeitung durch den Hauptprozeß 32 im Gegensatz zu der Last des Übertragens der überflüssigen Steuerbefehle über die Segmente zu den Teilprozessen 36, 37, muß entschieden werden, ob diese Optimierung implementiert wird oder nicht. Wenn der Hauptprozeß 32 kein Engpaß ist, kann der Hauptübersetzungszustand aus einer vollständigen Analyse jedes Steuerbefehls aufgebaut werden, und diese Optimierung verbessert den Gesamtdurchsatz des Systems.

Insofern als die unabhängigen PDL-Datenstromsegmente völlig unabhängig sein müssen, ist es notwendig, daß außerdem erweiterte Steuerbefehle in alle nachfolgenden unabhängigen PDL-Datenstromsegmente eingefügt werden. Die Einfügung erweiterter Steuerbefehle in diese nachfolgenden unabhängigen PDL-Datenstromsegmente hat drei wichtige Nachteile:

- 1) Solch ein erweiterter Steuerbefehl kann möglicherweise zu groß sein, um in mehr als ein

unabhängiges PDL-Datenstromsegment eingefügt zu werden.

5        2)        Die Verarbeitung erweiterter Steuerbefehle erfordert eine beträchtliche Verarbeitungszeit. Es sollte nicht allen Teilprozessen dieselbe Last auferlegt werden.

10       3)        Die aus einem solchen erweiterten Steuerbefehl erzeugten erweiterten Datenstrukturen können möglicherweise zu groß sein, um von jedem Teilprozeß getrennt gespeichert zu werden.

15       Bei einer bevorzugten Ausführungsform kann das erste und das zweite Problem gelöst werden, indem ein erweiterter Steuerbefehl nicht in ein unabhängiges PDL-Datenstromsegment eingefügt, sondern durch den Hauptprozeß 32 oder einen zentralen Prozeß verarbeitet wird. Das dritte Problem wird durch die zentrale  
20       Speicherung der aus dem erweiterten Steuerbefehl erzeugten erweiterten Datenstruktur gelöst. Die Nachteile dieser Lösung sind die folgenden:

25       1)        Sie würde zu einer zusätzlichen Belastung des Hauptprozesses 32 führen;

2)        es könnten nicht mehrere erweiterte Steuerbefehle gleichzeitig bearbeitet werden.

30       Bei einer besonders bevorzugten Ausführungsform wird das erste und das zweite Problem gelöst, indem jeder einzelne erweiterte Steuerbefehl nur in ein unabhängiges PDL-Datenstromsegment eingefügt wird. Durch diese Wahlmöglichkeit ist nur diesem Teilprozeß,  
35       durch den er verarbeitet wird, der erweiterte Steuerbefehl bewußt. Wenn mehrere Teilprozesse parallel ablaufen, ist es möglich, daß mehrere erweiterte Steuerbefehle, wie zum Beispiel die Definition oder

- Erzeugung von Fonts, gleichzeitig verarbeitet werden. Das dritte Problem wird gelöst, indem die erweiterte Datenstruktur zur Speicherung auf einem Speichergerät 46 zu einem zentralen Speicherprozeß 45 gesendet wird.
- 5 Vorzugsweise wird ein erweiterter Steuerbefehl in einem unabhängigen PDL-Datenstromsegment (d.h. einem "Dummy"-Segment) gespeichert, das darüber hinaus keine anderen erweiterten Steuerbefehle oder Datenbefehle enthält. Die Verarbeitung dieser Art von Segment führt zu einem
- 10 leeren Übergangs-Datenstromteil und zusätzlich zu der Erzeugung einer erweiterten Datenstruktur, die zur Speicherung auf dem Speichergerät 46 zu dem zentralen Speicherprozeß 45 gesendet wird.
- 15 Um andere Teilprozesse über die Existenz des erweiterten Steuerbefehls zu informieren, wird vorzugsweise ein spezifischer Steuerbefehl, der als der Steuerbefehl "version" bezeichnet wird, in alle nachfolgenden unabhängigen PDL-Datenstromsegmente
- 20 eingefügt. Ein Version-Steuerbefehl hat dieselbe Syntax wie gewöhnliche Steuerbefehle, wird aber nur durch den Hauptprozeß 32 eingeführt und in unabhängige PDL-Datenstromsegmente eingeführt. Der Version-Steuerbefehl hat außerdem die Funktion, die erweiterten Datenstruk-
- 25 turen, die aus dem erweiterten Steuerbefehl erzeugt werden, allen Teilprozessen zur Verfügung zu stellen. Jeder Teilprozeß führt vorzugsweise eine lokale Tabelle, in der gültige erweiterte Datenstrukturen identifiziert werden. Diese Identifikation wird zum
- 30 Zugriff auf die erweiterten Datenstrukturen über den zentralen Speicherprozeß 45 verwendet. Jeder Version-Steuerbefehl enthält eine solche Identifikation.
- Ein Version-Steuerbefehl wird für die Erzeugung und
- 35 Entfernung einer erweiterten Datenstruktur eingeführt. Bei der Erzeugung wird ein Eintrag in der lokalen Tabelle des Teilprozesses hinzugefügt, der die erzeugte erweiterte Datenstruktur identifiziert. Bei der

11.00.02

Entfernung wird der entsprechende Eintrag in der Tabelle entfernt. Es ist von äußerster Wichtigkeit, daß die Abfolge dieser Operationen ordnungsgemäß aufrechterhalten wird. Der Hauptprozeß 32 muß die richtige Abfolge beibehalten, so daß jeder Teilprozeß die Version-Steuerbefehle in der erforderlichen Reihenfolge erhält. Außerdem darf derselbe Teilprozeß nicht denselben Version-Steuerbefehl zwei- oder mehrmals vollständig verarbeiten und damit seine internen Tabellen aktualisieren.

Gemäß der obigen Besprechung der Steuerbefehlreduktion wird in den Kopfteil jedes unabhängigen PDL-Datenstromsegments ein "Version"-Steuerbefehl eingefügt, und zwar vorzugsweise so lange, bis alle Teilprozesse 36, 37 dem Hauptprozeß 32 die Verarbeitung des "Version"-Steuerbefehls zurückgemeldet haben.

Als Alternative führt der Hauptprozeß 32 eine Tabelle mit Identifikationen der gültigen erweiterten Datenstrukturen und sendet diese Tabelle über einen Kopfteil in jedem unabhängigen PDL-Datenstromsegment zu jedem Teilprozeß. Durch einen Antwortmechanismus, wie oben beschrieben, kann die Größe der über die unabhängigen PDL-Datenstromsegmente übermittelten Tabelle immer dann reduziert werden, wenn ein bestimmter gemeinsamer Tabellen-Level für alle Teilprozesse erreicht wird. Der Hauptprozeß 32 muß die Tabelle in derselben Abfolge aktualisieren in der sie aufgebaut wurde.

Um erweiterte Datenstrukturen abzurufen, die mit einem erweiterten Steuerbefehl zusammenhängen, muß jeder Teilprozeß auf den zentralen Speicherprozeß 45 zugreifen.

Außerdem kann beabsichtigt sein, daß ein Steuerbefehl in dem Eingangsdatenstrom eine erweiterte Datenstruktur entfernt oder überschreibt. Dazu kann es zum Beispiel

dann kommen, wenn ein spezifischer Font nicht mehr benutzt wird oder wenn ein Overlay durch ein anderes Overlay ersetzt werden muß usw.

- 5 Da eine erweiterte Datenstruktur durch Datenbefehle in einem nächsten unabhängigen PDL-Datenstromsegment S2 nach einem ersten unabhängigen PDL-Datenstromsegment S1, das den erweiterten Steuerbefehl enthält, referenziert werden kann und S2 möglicherweise von
- 10 einem anderen Teilprozeß P2 verarbeitet wird, als ein erster Teilprozeß P1, der S1 verarbeitet, muß ein Mechanismus Synchronisation bereitgestellt, damit sich der andere Teilprozeß P2 nicht auf die erweiterte Datenstruktur bezieht, bevor sie durch den ersten
- 15 Teilprozeß P1 erzeugt wurde. Andererseits darf ein Steuerbefehl in dem PDL-Datenstrom, der eine erweiterte Datenstruktur entfernen oder überschreiben soll, von keinem Teilprozeß verarbeitet werden, bevor andere Teilprozesse die Bearbeitung vorheriger unabhängiger
- 20 PDL-Datenstromsegmente vollständig abgeschlossen haben. Außerdem dürfen unabhängige PDL-Datenstromsegmente nach der Entfernung durch einen Steuerbefehl nicht verarbeitet werden, bevor die Entfernung wirksam wird.
- 25 Bei einer Ausführungsform wird, um die erforderliche Synchronisation bereitzustellen, ein "Segment"-Steuerbefehl eingeführt, der dieselbe Syntax hat wie gewöhnliche Steuerbefehle, aber nur vom Hauptprozeß 32 eingeführt und in unabhängige PDL-Datenstromsegmente
- 30 eingefügt wird. In jedes unabhängige PDL-Datenstromsegment wird nur ein Segmentsteuerbefehl eingefügt. Jeder Segmentsteuerbefehl erhält eine systemweit eindeutige Identifikation, die als Segment-ID bezeichnet wird. Der Hauptprozeß 32 signalisiert dem
- 35 zentralen Speicherprozeß 45 die Identifikation der erweiterten Datenstrukturen, die durch Datenbefehle referenziert werden, zusammen mit der Segment-ID des unabhängigen PDL-Datenstromsegments, in das diese

Datenbefehle eingefügt werden. Der Hauptprozeß 32 muß deshalb die Datenbefehle als Teil dieses unabhängigen PDL-Datenstromsegments analysieren. Außerdem signalisiert der Hauptprozeß 32 dem zentralen Speicherprozeß 45 die Entfernung oder das Überschreiben einer erweiterten Datenstruktur zusammen mit der Segment-ID des unabhängigen PDL-Datenstromsegments, in das der erweiterte Steuerbefehl eingefügt wird. Der Hauptprozeß 32 signalisiert dem zentralen Speicherprozeß 45 die Erzeugung einer erweiterten Datenstruktur zusammen mit der Segment-ID des unabhängigen PDL-Datenstromsegments, in das der erweiterte Steuerbefehl eingefügt wird. Immer wenn eine erweiterte Datenstruktur entweder für Entfernungszwecke oder für Erzeugungszwecke oder für Verwendungszwecke durch einen Datenbefehl durch einen Teilprozeß referenziert wird, muß die Segment-ID des unabhängigen PDL-Datenstromsegments, in das dieser erweiterte Steuerbefehl oder Datenbefehl eingefügt wird, dem zentralen Speicherprozeß 45 signalisiert werden. Dementsprechend kann der zentrale Speicherprozeß 45 die Benutzung der erweiterten Datenstrukturen in den verschiedenen unabhängigen PDL-Datenstromsegmenten mitverfolgen. Der zentrale Speicherprozeß 45 suspendiert die Entfernung oder das Überschreiben einer erweiterten Datenstruktur immer dann, wenn sie immer noch in vorherigen unabhängigen PDL-Datenstromsegmenten referenziert werden soll, und so lange, bis diese erweiterte Datenstruktur nicht mehr von Datenbefehlen in vorherigen unabhängigen PDL-Datenstromsegmenten referenziert wird. Der zentrale Speicherprozeß 45 suspendiert die Erzeugung einer erweiterten Datenstruktur immer dann, wenn eine vorherige Entfernung noch nicht wirksam ist. Der zentrale Speicherprozeß 45 suspendiert die Benutzung erweiterter Datenstrukturen in nachfolgenden unabhängigen PDL-Datenstromsegmenten, bis die erweiterten Datenstrukturen erzeugt wurden. Der Nachteil dieser Lösung besteht darin, daß der Hauptprozeß 32 die Datenbefehle



analysieren muß. Dadurch kann die Verarbeitungslast des Hauptprozesses 32 sehr groß werden.

Bei einer besonders bevorzugten Ausführungsform wird  
5 ein "sync"-Steuerbefehl eingeführt, der dieselbe Syntax  
hat wie gewöhnliche Steuerbefehle, aber nur vom  
Hauptprozeß 32 eingeführt wird und in unabhängige PDL-  
Datenstromsegmente eingefügt wird. Jeder sync-  
Steuerbefehl erhält eine systemweit eindeutige  
10 Identifikation, die als sync-ID bezeichnet wird. Durch  
Verarbeitung eines sync-Steuerbefehls in einem  
unabhängigen PDL-Datenstromsegment meldet jeder  
Teilprozeß den Empfang des sync an den Hauptprozeß 32  
zurück und benachrichtigt außerdem über die sync-ID.  
15 Eine sync-Steuerungsantwort kann auf dieselbe Weise wie  
eine Steuerbefehlantwort 44 erfolgen.

Ein sync-Steuerbefehl kann an einer beliebigen Stelle  
in einem unabhängigen PDL-Datenstromsegment eingefügt  
20 werden. Zum Beispiel kann ein sync-Steuerbefehl in den  
Kopfteil eines unabhängigen PDL-Datenstromsegments oder  
in den Nachspann eines unabhängigen PDL-  
Datenstromsegments eingefügt werden.

25 Bei einer bevorzugten Ausführungsform kann jedes  
unabhängige PDL-Datenstromsegment ein oder mehrere  
Teilsegmente enthalten. Alle Teilsegmente, die zu einem  
Segment gehören, werden durch denselben Teilprozeß  
verarbeitet. Obwohl jedes Segment oder eine Referenz  
30 darauf als Ganzes in die FIFO-Warteschlange 34  
eingereicht wird, können nachfolgende Teilsegmente zur  
durch den Hauptprozeß 32 einzelne Verarbeitung  
freigegeben werden. Durch diese Möglichkeit können  
mehrere Teilsegmente eines sehr großen unabhängigen  
35 PDL-Datenstromsegments erzeugt, in Puffern gespeichert  
und durch den Hauptprozeß 32 zur Verarbeitung  
freigegeben werden, sogar bis der Pufferraum erschöpft  
wird. Obwohl der Hauptprozeß 32 nicht alle

Teilsegmente, die zu dem sehr großen unabhängigen PDL-Datenstromsegment gehören, verarbeitet hat, verarbeitet der Teilprozeß, der das sehr große unabhängige PDL-Datenstromsegment ausgewählt hat, die freigegebenen  
5 Teilsegmente und gibt die Puffer der verarbeiteten Teilsegmente frei. Diese freigegebenen Puffer können dann von dem Hauptprozeß 32 für nachfolgende Teilsegmente des sehr großen unabhängigen PDL-Datenstromsegments verwendet werden.

10 Ein Teilprozeß, der über ein zu verarbeitendes unabhängiges PDL-Datenstromsegment einen sync-Steuerbefehl empfängt, muß den Hauptprozeß 32 sofort benachrichtigen, wenn dieser sync-Steuerbefehl von  
15 diesem Teilprozeß bearbeitet wird. Sobald ein unabhängiges PDL-Datenstromsegment, das einen solchen sync-Steuerbefehl als Teil eines Teilsegments enthält, in die FIFO-Warteschlange 34 eingereiht wird, können die nachfolgenden Teilsegmente und die nächsten  
20 unabhängigen PDL-Datenstromsegmente weiter durch den Hauptprozeß 32 erzeugt werden. Bei einer bevorzugten Ausführungsform wird ihre Freigabe zur Verarbeitung, die durch den Hauptprozeß 32 gegeben wird, verzögert, bis der Hauptprozeß 32 eine Antwort von dem sync-  
25 Steuerbefehl empfängt. Sobald die Antwort empfangen wird, gibt der Hauptprozeß 32 die nachfolgenden Teilsegmente und die unabhängigen PDL-Datenstromsegmente, die für die Verarbeitung vorbereitet wurden, zur Verarbeitung durch die Teilprozesse frei, mit  
30 Ausnahme von Teilsegmenten oder unabhängigen PDL-Datenstromsegmenten, die einem anderen sync-Steuerbefehl folgen. Solche Teilsegmente können aktiviert werden, wenn für diesen anderen sync-Steuerbefehl eine Antwort empfangen wird.

35 Als Alternative stoppt der Hauptprozeß 32 die Erzeugung neuer unabhängiger PDL-Datenstromsegmente und Teilsegmente, bis die sync-Steuerantwort empfangen wurde.

Ein sync-Steuerbefehl wird am Ende eines unabhängigen PDL-Datenstromsegments eingefügt, das einen erweiterten Steuerbefehl enthält. Ein Teilprozeß, der den Empfang eines solchen sync-Steuerbefehls an den Hauptprozeß 32 zurückmeldet, bestätigt folgendes:

a) daß der erweiterte Steuerbefehl verarbeitet wird, so daß sich eine erweiterte Datenstruktur ergibt; und

b) daß die erweiterte Datenstruktur allen Teilprozessen über den zentralen Speicherprozeß 45 verfügbar ist.

Erweiterte Steuerbefehle zur Entfernung oder zum Überschreiben erweiterter Datenstrukturen dürfen ebenfalls durch keinen Teilprozeß verarbeitet werden, bevor vorherige unabhängige PDL-Datenstromsegmente vollständig von anderen Teilprozessen verarbeitet wurden, oder zumindest diejenigen unabhängigen PDL-Datenstromsegmente, die die erweiterte Datenstruktur benutzen. Dementsprechend wird ein sync-Steuerbefehl vorzugsweise als ein Kopfteil-Teilsegment in ein unabhängiges PDL-Datenstromsegment eingefügt, das einen solchen erweiterten Steuerbefehl enthält. Der Hauptprozeß 32 gibt das Teilsegment, das den sync-Steuerbefehl enthält, zur Verarbeitung durch einen Teilprozeß frei und gibt keine weiteren Teilsegmente oder unabhängigen PDL-Datenstromsegmente frei. Bei einer bevorzugten Ausführungsform werden diese weiteren Teilsegmente und unabhängigen PDL-Datenstromsegmente von dem Hauptprozeß 32 erzeugt, aber nicht zur Verarbeitung freigegeben. Ein Teilprozeß, der das Teilsegment mit dem sync-Steuerbefehl verarbeitet, meldet den Empfang eines solchen sync-Steuerbefehls an einen Pufferabwicklungsprozeß oder direkt an den Hauptprozeß 32 zurück. Ein Abfolgemechanismus stellt

sicher, daß der Hauptprozeß 32 auf die sync-Steuerantwort reagiert, nachdem alle vorherigen unabhängigen PDL-Datenstromsegmente und Teilsegmente dieses unabhängigen PDL-Datenstromsegments, oder zumindest

5 diejenigen, die die erweiterten Datenstrukturen benutzen, verarbeitet wurden. Zu diesem Zeitpunkt gibt der Hauptprozeß 32 die nachfolgenden Teilsegmente zur Verarbeitung frei, so daß der erweiterte Steuerbefehl zum Entfernen oder Überschreiben erweiterter

10 Datenstrukturen durch den Teilprozeß verarbeitet werden kann, der den sync-Steuerbefehl abgewickelt hat. Es ist also klar, daß die sync-Steuerantwort letztendlich nur dann durch den Hauptprozeß 32 verarbeitet werden kann, wenn alle vorherigen unabhängigen PDL-

15 Datenstromsegmente durch die Teilprozesse 36, 37 verarbeitet wurden. Diese Konzeptanforderung wird unten weiter beschrieben.

Ein unabhängiges PDL-Datenstromsegment, das einen

20 erweiterten Steuerbefehl zum Entfernen oder Überschreiben einer erweiterten Datenstruktur enthält, erhält außerdem einen sync-Steuerbefehl in einem Abspann-Teilsegment des unabhängigen PDL-Datenstromsegments. Auf diese Weise wird die

25 Verarbeitung der nachfolgenden Teilsegmente und der nächsten unabhängigen PDL-Datenstromsegmente verzögert, bis die Entfernung tatsächlich bewirkt worden ist.

Wie oben beschrieben, erzeugt jeder Teilprozeß vier

30 Arten oder Kategorien von Antwortdaten:

1. Steuerbefehlantworten;
2. Version-Steuerantworten;
3. Fehlerantworten; und
- 35 4. sync-Steuerantworten.

Die Verkettung aller Antworten einer Kategorie führt zu einem Antwort-Datenstrom pro Antwort-Datentyp. Für

jedes unabhängige PDL-Datenstromsegment wird ein Teil jedes Antwortdatenstroms erzeugt, der im folgenden als ein Antwortdatenteil bezeichnet wird. Wegen der Beschaffenheit der Antwortdaten ist es möglich, daß bestimmte Antwortdatenteile leer sind. Wegen der Parallelität der Teilprozesse 36, 37 ist es möglich, daß Antwortdatenteile für den entsprechenden Antwortdatenstrom nicht in derselben chronologischen Reihenfolge vorliegen, in der die entsprechenden unabhängigen PDL-Datenstromsegmente von dem Hauptprozeß 32 erzeugt wurden. Für Fehlerantworten ist es eine Systemanforderung, daß die Fehlerantwortteile zu einem Fehlerantwortdatenstrom kombiniert werden, so wie es durch die Reihenfolge des PDL-Datenstroms 31 auferlegt wird. Bei den sync-Steuerantworten besteht eine Konzeptanforderung darin, daß die sync-Steuerantwortteile zu einem sync-Steuerantwortdatenstrom kombiniert werden, so wie es durch die Reihenfolge der erzeugten unabhängigen PDL-Datenstromsegmente auferlegt wird. Es ist vorzuziehen, daß die vier Antwortdatenteile - Steuerbefehlantwort, Version-Steuerantwort, sync-Steuerantwort und Fehlerantwort - von dem Teilprozeß für jedes verarbeitete unabhängige PDL-Datenstromsegment zu einem Antwortdatenteil kombiniert werden. Der Hauptprozeß 32 nimmt diese Teile von den Teilprozessen 36, 37 an und ordnet sie in der richtigen Abfolge um.

Außerdem wurde bereits beschrieben, daß jeder Teilprozeß einen Übergangs-Datenstromteil 38 erzeugt, der effizienter in eine Bitmap-Darstellung übersetzt werden kann. Für jedes unabhängige PDL-Datenstromsegment, das Datenbefehle enthält, wird ein Teil des Übergangs-Datenstroms erzeugt. Wegen der Beschaffenheit des unabhängigen PDL-Datenstromsegments (z.B. eines "Dummy"-Segments) ist es möglich, daß bestimmte Übergangs-Datenstromteile leer sind. Wegen der Parallelität der Teilprozesse 36, 37 ist es

möglich, daß Übergangs-Datenstromteile nicht bereits in derselben chronologischen Reihenfolge vorliegen, in der die entsprechenden unabhängigen PDL-Datenstromsegmente durch den Hauptprozeß 32 erzeugt wurden. Es ist eine Systemanforderung, daß diese Teile in der Reihenfolge, die durch die Reihenfolge des PDL-Datenstroms auferlegt wird, zu einem Übergangs-Datenstrom 40 kombiniert werden. Ein Kombinationsprozeß 39, der die Übergangs-Datenstromteile von den Teilprozessen 36 oder 37 annimmt, ordnet diese Teile in die richtige Abfolge um.

Bei einer ersten Ausführungsform nutzt der Kombinationsprozeß 39 Abfolgeinformationen aus, die ursprünglich durch den Hauptprozeß 32 erzeugt wurden. Bei einer ersten Ausführungsform kann der Hauptprozeß 32 jedes Mal, wenn er ein neues unabhängiges PDL-Datenstromsegment erzeugt, dem unabhängigen PDL-Datenstromsegment eine eindeutige, sich erhöhende Abfolgenummer zuweisen. Der Teilprozeß, der dieses unabhängige PDL-Datenstromsegment verarbeitet, leitet die Abfolgenummer zu dem Übergangs-Datenstromteil weiter. Der Kombinationsprozeß 39 entnimmt beim Empfang des unabhängigen PDL-Datenstromteils die Abfolgenummer und verwendet sie, um die ankommenden Übergangs-Datenstromteile in die richtige Reihenfolge umzuordnen. Die Abfolgenummern können außerdem zum Umordnen von Steuerbefehlantworten, Version-Steuerantworten, sync-Steuerantworten und Fehlerantworten dienen. Der Teilprozeß, der ein unabhängiges PDL-Datenstromsegment verarbeitet, kann die Abfolgenummer in den Antwortdatenteil einfügen. Der Hauptprozeß 32 entnimmt beim Empfang des Antwortdatenteils die Abfolgenummer und verwendet sie, um die ankommenden Antwortdatenteile in die richtige Reihenfolge umzuordnen.

Bei einer zweiten Ausführungsform teilt ein separater Pufferabwicklungsprozeß, der Teil des Hauptprozesses 32 sein kann, drei Zeiger für jedes unabhängige PDL-

Datenstromsegment zu. Ein erster Zeiger bezieht sich auf das unabhängige PDL-Datenstromsegment. Ein zweiter Zeiger bezieht sich auf den Antwortdatenteil, der von dem Teilprozeß erzeugt wird, der das unabhängige PDL-Datenstromsegment verarbeitet. Ein dritter Zeiger bezieht sich auf den Übergangs-Datenstromteil, der aus dem unabhängigen PDL-Datenstromsegment erzeugt wird. Jedesmal wenn ein Teilprozeß ein unabhängiges PDL-Datenstromsegment zur Verarbeitung zuteilt, erhält er diese drei Zeiger und verarbeitet das unabhängige PDL-Datenstromsegment. Bei Beendigung des unabhängigen PDL-Datenstromsegments durch den Teilprozeß sendet ein Pufferabwicklungsprozeß den Zeiger für den Übergangs-Datenstromteil immer dann zu dem Kombinationsprozeß, wenn dieser Zeiger dem nächsten Zeiger aus der FIFO-Warteschlange 34 entspricht. Der Pufferabwicklungsprozeß sendet außerdem den Zeiger für den Antwortdatenteil immer dann zu dem Hauptprozeß 32, wenn dieser Zeiger dem nächsten Zeiger aus der FIFO-Warteschlange 34 entspricht. Bei Beendigung der Verarbeitung dieses unabhängigen PDL-Datenstromsegments signalisiert der Teilprozeß diesen Umstand dem Pufferabwicklungsprozeß, der ihn wiederum dem Hauptprozeß 32 und dem Kombinationsprozeß 39 signalisiert. Vorzugsweise ist der Pufferabwicklungsprozeß ein separater Prozeß, so daß seine Last von dem Hauptdatenstrom ferngehalten wird, der einen Engpaß bilden kann.

Der Kombinationsprozeß bietet den resultierenden umgeordneten Übergangs-Datenstrom 40 einem Bitmap-Erzeugungsprozeß 41 an, der eine Bitmap-Darstellung erzeugt. Die Bitmap-Darstellung wird zu einer Markierungsmaschine 43 gesendet, die das durch die Bitmap-Darstellung dargestellte Bild auf die Druckausgabe schreibt, bei der es sich um Papier, fotografischen Film, ein wärmeempfindliches Medium usw. handeln kann. Sobald ein latentes Bild auf einem fotografischen Film oder Medium gebildet wurde, muß das

Medium durch in der Technik bekannte Verfahren chemisch verarbeitet und entwickelt werden.

- 5 Nach der ausführlichen Beschreibung bevorzugter Ausführungsformen der vorliegenden Erfindung ist für Fachleute nunmehr erkennbar, daß zahlreiche Modifikationen vorgenommen werden können, ohne den Schutzzumfang der Erfindung zu verlassen, der in den folgenden Ansprüchen definiert wird.



GN95002

- 32 -

## ANSPRÜCHE:

1. Verfahren zur Wiedergabe eines Bildes, das durch einen PDL-Datenstrom beschrieben wird, mit den folgenden Schritten:
  - Erkennen von Datenbefehlen und Steuerbefehlen in dem PDL-Datenstrom;
  - Erzeugen mehrerer unabhängiger PDL-Datenstrom-segmente;
  - Zuweisen jedes Datenbefehls zu einem der Segmente;
  - Zuweisen von Steuerbefehlen zu den Segmenten;
  - Übersetzen jedes Segments in Übergangs-Datenstromteile;
  - Kombinieren der Teile zu einem Übergangs-Datenstrom in einer Reihenfolge, die durch den PDL-Datenstrom auferlegt wird; und
  - Wiedergeben des Bildes aus dem Übergangs-Datenstrom,
- 20 dadurch gekennzeichnet, daß mindestens einer der erkannten Steuerbefehle mehr als einem der unabhängigen PDL-Stromsegmente zugewiesen wird.
2. Verfahren nach Anspruch 1, weiterhin mit den folgenden Schritten:
  - Ableiten eines Übersetzungszustands von den erkannten Steuerbefehlen;
  - Erzeugen von Steuerbefehlen für jedes Segment, die den Übersetzungszustand beschreiben.
- 30 3. Verfahren nach Anspruch 1, wobei
  - die Segmente durch einen Hauptprozeß erzeugt werden;
  - die Segmente von einem oder mehreren Teilprozessen übersetzt werden;
  - jeder Teilprozeß Segmente zur Verarbeitung auswählt.

4. Verfahren nach Anspruch 3, wobei
- der Hauptprozeß einen Hauptübersetzungszustand von den Steuerbefehlen in dem PDL-Datenstrom ableitet;
  - 5        - jeder Teilprozeß einen Teilprozeß-Übersetzungszustand von den Steuerbefehlen in den unabhängigen PDL-Datenstromsegmenten ableitet;
  - 10       - jeder Teilprozeß eine Änderung des Teilprozeß-Übersetzungszustands durch eine Steuerbefehlantwort zu dem Hauptprozeß zurückgibt.
5. Verfahren nach Anspruch 4, wobei der Hauptprozeß
- 15       - einen gemeinsamen Übersetzungszustands-Level aus den Steuerbefehlantworten berechnet;
  - für jedes Segment Steuerbefehle erzeugt, die den Unterschied zwischen dem Hauptübersetzungszustand und dem gemeinsamen Übersetzungszustands-Level beschreiben.
  - 20
6. Verfahren nach Anspruch 4, wobei der Hauptprozeß
- den Inhalt von Steuerbefehlen analysiert;
  - überflüssige Steuerbefehle aus den Segmenten zurückweist.
  - 25
7. Verfahren nach Anspruch 1, weiterhin mit den folgenden Schritten:
- Erkennen von reduzierten Steuerbefehlen und erweiterten Steuerbefehlen in den Steuerbefehlen;
  - 30       - Zuweisen jedes erweiterten Steuerbefehls zu höchstens einem Segment;
  - Erzeugen eines Version-Steuerbefehls für mindestens ein Segment, der den erweiterten Steuerbefehl angibt.
  - 35
8. Verfahren nach Anspruch 7, weiterhin mit den folgenden Schritten:

110000

- Erzeugen eines sync-Steuerbefehls in dem Segment, das den erweiterten Steuerbefehl enthält;
  - Verzögern der Übersetzung nachfolgender Segmente, bis eine sync-Steuerantwort aus dem sync-Steuerbefehl empfangen wurde.
- 5
9. Verfahren nach Anspruch 8, wobei die sync-Steuerantwort erst dann empfangen wird, wenn alle vorherigen Segmente verarbeitet wurden.
- 10
10. Verfahren nach Anspruch 7, weiterhin mit den folgenden Schritten:
- Erzeugen eines Segment-Steuerbefehls in jedem Segment;
  - Erkennen von Befehlen zum Entfernen einer erweiterten Datenstruktur unter den erweiterten Steuerbefehlen;
  - Suspendieren des Entfernens der erweiterten Datenstruktur, bis für mindestens alle vorherigen Segmente, die die erweiterte Datenstruktur benutzen, für den Segment-Steuerbefehl eine Antwort empfangen wurde.
- 15
- 20
- 25 11. Verfahren nach Anspruch 10, weiterhin mit den folgenden Schritten:
- Erkennen von Befehlen zum Erzeugen einer erweiterten Datenstruktur unter den erweiterten Steuerbefehlen;
  - Suspendieren des Erzeugens der erweiterten Datenstruktur, bis ein vorhergehendes Entfernen der erweiterten Datenstruktur abgeschlossen wurde.
- 30
- 35 12. Verfahren nach Anspruch 11, weiterhin mit den folgenden Schritten:

110202

- Erkennen von Befehlen, die eine erweiterte Datenstruktur benutzen, unter den Datenbefehlen;
- Suspendieren der Benutzung der erweiterten Datenstruktur, bis die erweiterte Datenstruktur erzeugt wurde.

13. Bildwiedergabesystem, umfassend:

- ein Mittel zum Erkennen von Datenbefehlen und Steuerbefehlen in einem PDL-Datenstrom;
- ein Mittel zum Erzeugen mehrerer unabhängiger PDL-Datenstromsegmente;
- ein Mittel zum Zuweisen jedes Datenbefehls zu einem der Segmente;
- ein Mittel zum Zuweisen von Steuerbefehlen zu den Segmenten;
- ein Mittel zum Übersetzen jedes Segments in Übergangs-Datenstromteile;
- ein Mittel zum Kombinieren der Teile zu einem Übergangs-Datenstrom in einer Reihenfolge, die durch den PDL-Datenstrom auferlegt wird; und
- ein Mittel zum Wiedergeben eines Bildes aus dem Übergangs-Datenstrom;

wobei das Mittel zum Zuweisen von Steuerbefehlen zu den Segmenten dazu dient, mindestens einen Steuerbefehl mehr als einem der Segmente zuzuweisen.

14. Bildwiedergabesystem nach Anspruch 13 mit mindestens zwei Prozessoren, die jeweils ein unabhängiges Speichermittel aufweisen.

110302

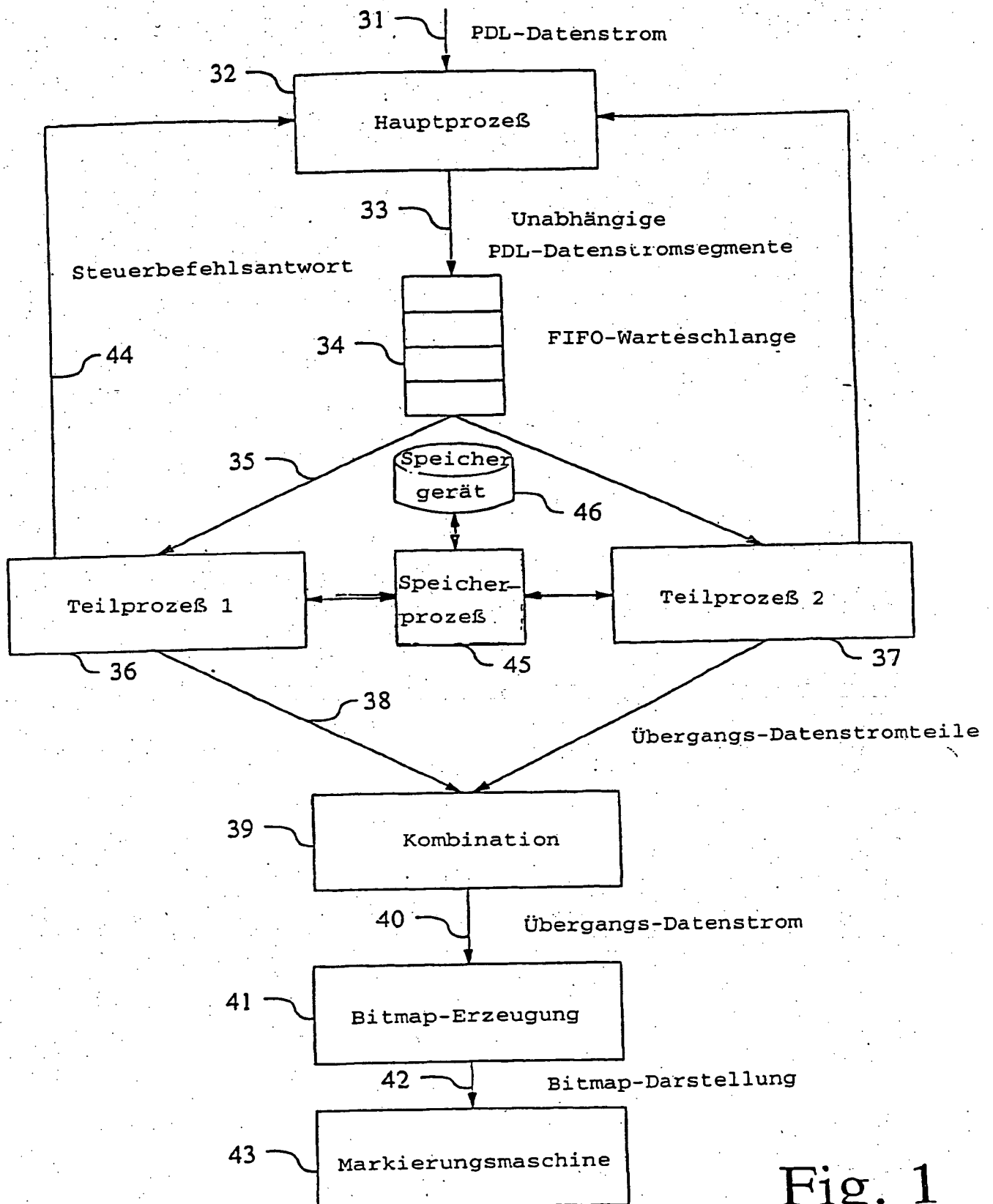


Fig. 1

1/9/1

DIALOG(R)File 351:Derwent WPI  
(c) 2003 Thomson Derwent. All rts. reserv.

010930310 \*\*Image available\*\*

WPI Acc No: 1996-427260/199643

XRPX Acc No: N96-359713

**Reproducing image described by PDL data stream - by segmenting original data stream and inserting control commands such that translation is handled by several processors in parallel**

Patent Assignee: AGFA-GEVAERT NV (GEVA )

Inventor: VENNEKENS P

Number of Countries: 007 Number of Patents: 006

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 733965	A1	19960925	EP 95200724	A	19950323	199643 B
JP 8297560	A	19961112	JP 9694779	A	19960325	199704
US 5652711	A	19970729	US 96618538	A	19960319	199736
JP 3194351	B2	20010730	JP 9694779	A	19960325	200146
EP 733965	B1	20011205	EP 95200724	A	19950323	200203
DE 69524381	E	20020117	DE 624381	A	19950323	200213
			EP 95200724	A	19950323	

Priority Applications (No Type Date): EP 95200724 A 19950323

Cited Patents: EP 574224; WO 9411805

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
-----------	------	-----	----	----------	--------------

EP 733965	A1	E	12	G06F-003/12	
-----------	----	---	----	-------------	--

Designated States (Regional): BE DE FR GB NL

JP 8297560	A		13	G06F-003/153	
------------	---	--	----	--------------	--

US 5652711	A		10	G06T-009/00	
------------	---	--	----	-------------	--

JP 3194351	B2		12	G06F-003/12	Previous Publ. patent JP 8297560
------------	----	--	----	-------------	----------------------------------

EP 733965	B1	E		G06F-003/12	
-----------	----	---	--	-------------	--

Designated States (Regional): BE DE FR GB NL

DE 69524381	E			G06F-003/12	Based on patent EP 733965
-------------	---	--	--	-------------	---------------------------

Abstract (Basic): EP 733965 A

The method involves recognising data commands and control commands in a data stream. Several independent stream segments are generated. Each data command is assigned to one of the segments. A control command is assigned to more than one of the segments.

Each segment is translated into intermediate data stream portions which are combined into one intermediate data stream. The image is reproduced from the intermediate data stream. A translation state is derived from the control commands. Control commands describing the translation state are generated for each segment.

USE/ADVANTAGE - For producing hard copy on paper, thermosensitive material or photographic film. For desktop publishing. For video monitor. Increases processor performance. Allows several processors to operate simultaneously. Linear performance. Maintains correct sequence of pages when parallel processing.

Dwg.1/1

Abstract (Equivalent): US 5652711 A

A method for reproducing an image described by a PDL data stream, comprising the following steps:

- recognising in said PDL data stream data commands and control commands;
- generating a plurality of independent PDL data stream segments;
- assigning each data command to one of said segments;
- assigning at least one control command to more than one of said segments;
- translating each of said segments to intermediate data stream portions;
- combining said portions to one intermediate data stream; and,
- reproducing said image from said intermediate data stream.

. Dwg.1/1

Title Terms: REPRODUCE; IMAGE; DESCRIBE; DATA; STREAM; SEGMENT; ORIGINAL; DATA; STREAM; INSERT; CONTROL; COMMAND; TRANSLATION; HANDLE; PROCESSOR; PARALLEL

Derwent Class: P75; T01

International Patent Class (Main): G06F-003/12; G06F-003/153; G06T-009/00

International Patent Class (Additional): B41J-029/38; G06T-001/20;

G06T-011/00; H04N-001/00

File Segment: EPI; EngPI

Manual Codes (EPI/S-X): T01-C05A; T01-J11